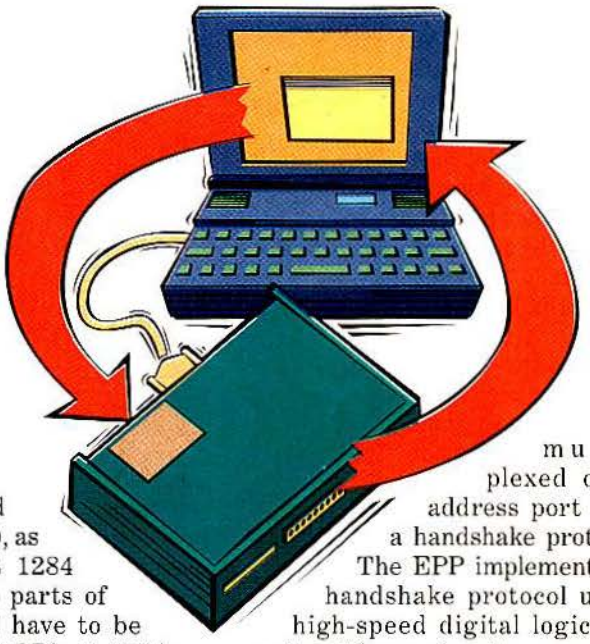
Shake Hands with the EPP



Allen Tracht,
IOtech, Cleveland, OH

The Enhanced Parallel Port's handshaking provides two-way communication between PCs and data acquisition instruments. Using the EPP makes it easier to acquire data in the field, but the protocol currently works only under DOS.

Notebook PCs are ideal for acquiring data in the field. Just carry along a battery pack and a separate box to hold your instrument cards, and you can perform data acquisition anywhere. The only problem is that some notebook PCs lack internal industry-standard (ISA) expansion slots, and the RS-232 port is too slow for many data acquisition applications.

To circumvent this problem, some engineers rely on the PC's parallel port to communicate with external devices. The standard parallel port (SPP), however, has its own disadvantages for data acquisition applications: its 150 kB/s transfer rate is still too slow, and there is no standard for receiving data from external devices.

A better solution appears to be the enhanced parallel port (EPP), as defined in IEEE 1284 (draft).[1] Although parts of the standard still have to be ironed out, the EPP's 2 MB/s transfer rate is fast enough for most data acquisition applications, and the EPP is already available on some notebook PCs. By understanding how the EPP works, you'll be able to use it to improve the speed of your data acquisition applications.

Single Transfer Sequence Speeds Data

The EPP standard specifies different names for each of the parallel-port signal lines, one each for the SPP and EPP modes. As a result, the EPP is compatible with the SPP. During SPP operation, the lines are referred to by names that reflect the conventional printer port usage. During EPP operation, however, the signals are referred to by names that reflect the enhanced protocol (**Table 1**).

Currently, the EPP standard defines a general-purpose, 8-bit multiplexed data/address port with a handshake protocol. The EPP implements its handshake protocol using high-speed digital logic circuits. These circuits generate control strobes and handshake signals for transferring data to your instruments. The EPP needs only one CPU I/O cycle to complete an interlocked data transfer, and it can transfer data at 2 MB/s.

The EPP's handshake protocol is similar to that of the IEEE 488 bus, ensuring data integrity and protecting against data loss. Peripheral devices can accept data at a rate that matches their abilities to process or store it, even if that rate is less than 2 MB/s.

The benefits of the EPP become clear when you compare it to the SPP. To output 1 byte of data to a standard parallel printer, the SPP transfer sequence requires four I/O cycles and CPU instructions. The amount of time required to process the SPP's four I/O cycles limits the SPP's maximum transfer rate to 150 kB/s.

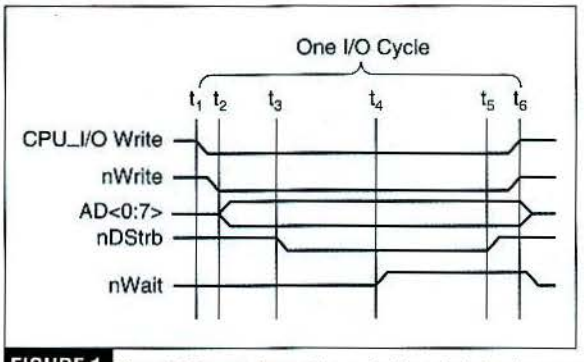


FIGURE 1. The EPP can transfer a byte of data in one I/O cycle (see text).

The Handshaking Sequence

Figure 1 shows the EPP handshaking sequence for writing data to a peripheral. At time t_1, the application program starts an EPP I/O write cycle by writing to the EPP's data port. The data port's address is usually 4 higher than the EPP's base address. For example, if the base address is 0278H (LPT1), then the EPP's data port is typically at 027CH.

At t_2, the host PC asserts nWrite by setting it low. At that time, the host places data on the data lines AD<0:7>. If the host detects that nWait is asserted (low), then nDStrb strobe will go low at t_3 and data transfer will begin. When the peripheral has accepted the data, it sets nWait high at t_4. At t_5, the host then sets nDStrb high in response to nWait being set high. The host ends the I/O cycle by setting nWrite and CPU_I/O Write high at t_6. The next cycle can begin after the peripheral asserts nWait by setting it low again.

The EPP protocol defines four types of transfer cycles:

- Data Write Cycle
- Data Read Cycle
- Address Write Cycle
- Address Read Cycle

Address and data cycles differ only in terms of which strobe the port asserts, nAStrb or nDStrb (**Table 1**). Address write cycles send address information to the EPP's registers while address read cycles receive identification information from the registers. Data read and write cycles transfer commands and test results between the PC and peripheral device. Typical EPP hardware implementations generate these I/O cycles by reading from or writing to the appropriate EPP interface register. **Table 2** lists the EPP software registers.

The EPP is also hardware independent. Rather than having to write to the printer-port hardware directly, you can obtain access to the EPP through a device driver, much like you would when communicating with an ISA-bus board.

The EPP BIOS, as defined by the IEEE P1284.3 committee, provides hardware independence by acting as a common software interface between your instrument driver and the EPP hardware. **Table 3** lists the EPP's BIOS calls that your driver must use.

PCs equipped with the EPP will have these functions embedded into their BIOS. Instrument suppliers must still provide you with instrument drivers for controlling your EPP-based instrument. Your test application program must supply parameters such as channel and gain to the instrument drivers for controlling the instrument.

All EPP BIOS calls return a status byte that an instrument driver can retrieve and send to your application program. You can use the status bytes (see **Table 4**) for error detection.

The BIOS includes a software module that supports multiplexed and daisy-chained peripherals. The module lets you attach multiple EPP peripherals to a single EPP port. The multiplexer (mux) and daisy chain calls are fielded by either an EPP mux-device driver or daisy-chain device driver. If neither of these drivers are present, the EPP BIOS will return the "Multiplexer not present" error code (40). A peripheral can work in either the multiplexed or daisy-chained environments.

Daisy-chain and mux arrangements are similar. A multiplexed device has a single port that connects it to the mux. A daisy-chain

TABLE 1. EPP/SPP Signal Definitions and DB-25 Pinouts

SPP Signal	EPP Signal	I/O	EPP Description	DB-25 Pin
nStrobe	nWrite	O	Active low. Indicates a write operation.	1
nAutoFd	nDStrb	O	Active low. Indicates a data read or write operation is in process.	14
nSelectIn	nAStrb	O	Active low. Indicates an Address read or write operation is in process.	17
nInit	nInit	O	Active low. Peripheral reset.	16
nACK	Intr	I	Peripheral interrupt. Generates an interrupt on the low to high transition	10
Busy	nWait	I	Handshake signal. When low it indicates that it is OK to start a cycle (assert a strobe). When high, it indicates that it is OK to end the cycle (deassert a strobe).	11
D<0:7>	AD<0:7>	BI	Bidirectional address/data lines.	2–9
PError	user defined	I		12
Select	user defined	I		13
nError	user defined	I		15

TABLE 2. EPP/SPP Software Register Definitions

Port Name	Offset*	Mode	Type	Description
SPP Data	0	SPP	Write	Standard SPP data port. No autostrobing.
SPP Status	1	SPP	Read	Reads the input status lines on the interface.
SPP Control	2	SPP	Write	Sets the state of the output control lines.
EPP Address	3	EPP	Read/Write	Generates an interlocked address read or write cycle.
EPP Data	4	EPP	Read/Write	Generates an interlocked data read or write cycle.
Not Defined	5–7			Used differently by various implementations.

*Offset from base address.

RATE THIS ARTICLE
Please let us know whether this article was:
Very Useful.....Circle 227
Somewhat Useful.....Circle 228
Not Useful.....Circle 229
Circle the corresponding number on the reader service card in this issue.

peripheral, however, has both input and output ports. A daisy-chained instrument's input port is connected to either the host PC's parallel port or to the daisy-chained peripheral in front of it. The output port is connected to the next peripheral in the daisy chain.

Still for DOS Only

Currently, the EPP BIOS standard provides interfaces only for DOS-based applications and device drivers. The IEEE P1284.3 subcommittee is, however, developing driver interfaces for Windows, Windows NT, OS/2, UNIX, and Apple-based computers. The IEEE 1284 specification includes platform-independent device drivers that will let engineers use the EPP on many different computers.

TABLE 3. Currently Defined EPP BIOS Calls

BIOS Call	Definition
Installation_Check	Tests for the presence of an EPP port.
Query_Config	Returns the EPP port's configuration.
Set_Mode	Sets the operating mode of the EPP port.
Get_Mode	Returns the current operating mode of the port.
Interrupt_Control	Enables or disables the interrupt for the EPP port.
EPP_Reset	Resets the peripheral device connected to the port.
Address_Write	Performs an address write cycle.
Address_Read	Performs an address read cycle.
Write_Byte	Outputs a single byte using a data write cycle.
Write_Block	Outputs a block of bytes using multiple data write cycles.
Read_Byte	Inputs a single byte using a data read cycle.
Read_Block	Inputs a block of bytes using multiple data read cycles.
Address/Byte_Read	Performs address write cycle, followed by a data read cycle.
Address/Byte_Write	Performs address write cycle, followed by a data write cycle.
Address/Block_Read	Performs address write cycle, followed by multiple data read cycles.
Address/Block_Write	Performs address write cycle, followed by multiple data write cycles.
Lock_Port	Locks an external multiplexer port when used in a multiport environment.
Unlock_Port	Unlocks, or releases, an external multiplexer port when used in a multiport environment.
Device_Interrupt	Allows EPP driver to install an interrupt handler for a PC device.
Set_Port_ID	Maps an EPP product ID onto a particular EPP or multiplexor port.
Set_Product_ID	Links a device driver with a peripheral attached to the EPP port.
Query_MUX	Returns state information about the EPP multiplexor.
Query_Device_Port	Returns state information about a particular multiplexor device port; can only be used independent of the lock status of the EPP channel.
Query_Daisy_Chain	Returns information about the EPP daisy chain.
Rescan_Daisy_Chain	Dynamically reassigns port numbers (1–8) to devices connected via the daisy chain; useful for daisy-chained devices that do not normally remain connected to the port.

Is the EPP the Best Choice for You?

The EPP represents an advance in PC connectivity comparable to the performance advances in the PC itself. For many applications, the EPP provides a communications alternative to serial or IEEE 488 in terms of flexibility, speed, and cost.

The EPP shows its most promise when used with notebook PCs, because notebook PCs lack the ISA bus needed for most data acquisition cards. A number of options exist for connecting instruments to PCs, any of which may be best for your application. Consider your present and future needs before committing to any configuration. Besides using the EPP, you can use

- a data acquisition instrument or a datalogger attached through an RS-232 port
- an instrument attached through the SPP
- an instrument attached through a GPIB port
- an ISA bus data acquisition plug-in board or an ISA bus expansion chassis with data acquisition boards
- a PCMCIA-compatible data acquisition card (plugged into a PCMCIA socket)

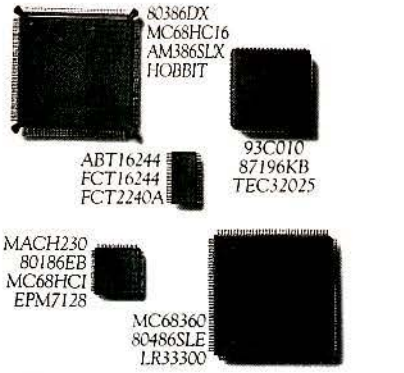
Here's an analysis of each option:

The RS-232 port. The most common method of connecting equipment to a PC is through its RS-232 serial data port. This universally available port supports data communication at rates up to about 10 kB/s or 20 kB/s. Because each reading requires two or more bytes of data, the sample rate is one-half of the raw data transfer rate. Therefore, RS-232-ported equipment is suitable only for applications that can tolerate a sample rate of 5-10 ksamples/s.

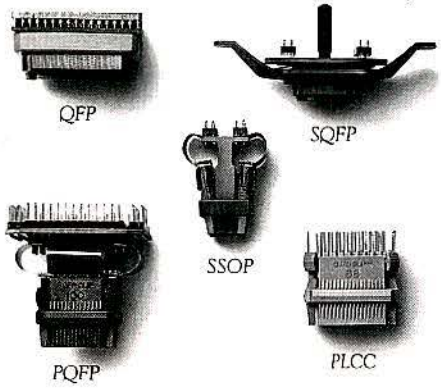
The standard parallel port (SPP). Today, the SPP can be used as an instrument interface for bi-directional data transfers at rates up to 150 kB/s. This is 10X the rate of the RS-232 serial interface, but

TABLE 4. Currently Defined EPP BIOS Error Codes

Error Code (hexadecimal)	Description
00	Successful.
01	Command/feature not supported.
02	Parameter error.
03	Unrecognized I/O base address.
80	I/O timeout (bit 7).
40	Multiplexor not present.
41	Multiplexor currently locked.
42	Port not currently locked.
43	Lock failure.



FOR THOSE OF YOU WHO LIVE BY THE NUMBERS,



HERE'S THE EASIEST WAY TO CHECK EVERYTHING OUT.

THE WORLD'S LARGEST OFFERING OF TEST CLIPS.

Today's boards contain chips that are too expensive to toss if they don't work. You could spend hours (or days) jury-rigging to access each lead because clock speeds and other functions make seeing everything essential.

Fortunately, Pomona offers a series of clips that allow you to test the newest chips on the market. And a FREE 1994 New Products Catalog makes finding the right test clip easy. Just look up the chip configuration or the device number, and you'll find the right Pomona test clip. You can also call direct for technical assistance at (909) 469-2912. Then call or visit your Pomona distributor for the clip you need. ITT Pomona Electronics, 1500 E. Ninth Street, Pomona, CA 91766-3835. (909) 469-2900. Fax (909) 629-3317.

ITT Pomona Electronics
AN ITT CANNON COMPANY

slower than that available with EPP, GPIB, and PCMCIA.

IEEE 488 (GPIB). IEEE 488 instruments are common in benchtop applications, but their bulk and their power requirement usually preclude them from portable applications. Notebook PCs, however, can control IEEE 488 instruments through printer-port-based IEEE 488 interfaces.

ISA bus plug-in cards. The majority of current PC-based data acquisition systems use ISA bus plug-in cards. Most are desktop systems, and the number of expansion slots is limited. If you need more slots, you can connect an expansion chassis to your system and add from two to eight ISA-bus slots. Of course, such systems are impractical for portable operations because of their bulk and their reliance on AC line power.

Once you establish the requirements for your data acquisition system, you can decide whether the EPP is best for you.

PCMCIA slots and boards. If portability is not an issue, ISA bus slots will accept plug-in cards from several manufacturers that offer a wide array of functions. The maximum data transfer rate is in the range of 1-3 MB/s.

The Personal Computer Memory Card Industry Association (PCMCIA) has been developing a standard for connecting credit-card size peripherals to PCs. Originally, the PCMCIA standard covered only memory cards, but it has been extended to I/O devices. A 68-pin connector is used for this interface, and data can be transferred at rates up to 5 MB/s.

PCMCIA cards, however, do not support direct memory access, which is available to ISA bus devices. This increases the time between data acquisition and CPU processing, so PCMCIA I/O cards currently require significant buffer memory. Proposed changes to the standard will help overcome this limitation.

Another disadvantage of PCMCIA devices is their small size. With maximum dimensions of 86x54x10 mm, they can't handle some of the analog components used in data acquisition circuits.

The limited availability of the PCMCIA interface is another disadvantage. Most portable PCs have only one or two PCMCIA sockets. PCMCIA cards have at most eight differential analog channels each. If you need more than 16 channels, PCMCIA data acquisition cards will not help. Most desktop computers don't have PCMCIA slots, making them incompatible if you want to use the same equipment on both a desktop and portable PC. You can, however, install a PCMCIA card slot in your desktop PC.

Before deciding which option to use, evaluate both what you need the system to do and the conditions under which you will use it. Answer these questions before selecting a system:

- How fast must the acquisition rate be?
- Will you want to analyze the data after acquiring it, or will you need it "on the fly"?
- Will you need two-way communication that ensures data integrity?
- What is the size and weight of your PC and data acquisition instruments?
- What resolution and data transfer rate will you need?
- How much power will your system need, and what type of power is available where you will use the system?

Once you establish the requirements for your data acquisition system, you can decide whether the EPP is best for you or whether you should use some other system. *T&MW*

Allen Tracht *is principal engineer at IOtech where he designed the EPP interface for the company's portable data acquisition systems. He holds a B.S. in Physics from the Massachusetts Institute of Technology and an M.S.E.E. from the California Institute of Technology. IOtech, Cleveland, OH. 216-439-4091.*

REFERENCE

1. IEEE STD 1284.3-1994, "Standard Signaling Method for a Bi-directional Parallel Peripheral Interface for Personal Computers" (draft). *IEEE 1284 Committee, Piscataway, NJ. 800-678-4333.*

For quantity reprints of this article, contact Ellen Sandkam at 708-390-2773.